UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/809,749	03/25/2004	Brian Robert Prasky	POU920030065US1	7321

33558 7590 10/30/2007

INTERNATIONAL BUSINESS MACHINES CORPORATION
IPLAW DEPARTMENT
2455 SOUTH ROAD - MS P386
POUGHKEEPSIE, NY 12601

EXAMINER	
FENNEMA, ROBERT E	
ART UNIT	**PAPER NUMBER**
2183	

MAIL DATE	DELIVERY MODE
10/30/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/809,749	PRASKY ET AL.	
	Examiner	Art Unit	
	Robert E. Fennema	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 September 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-8,10-18,20-28 and 30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8,10-18,20-28 and 30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-8, 10-18, 20-28, and 30 have been considered. Claims 1, 11, and 21 amended as per Applicant's request.

Claim Objections

2. In Claims 1, 11, and 21, the language which was added to the claims is not entirely clear, due to the use of multiple "ands", in the phrase "prevent the branch from being written into the branch history buffer and branch target buffer and predicted", which is unclear. Examiner strongly suggests changing the phrasing to "and preventing the branch from being predicted" instead of "and predicted" (or something along those lines to clarify), because it is not entirely clear if the Applicant has changed the scope of the claim such that branches, previously not predicted in the previous sets of claims, now does predict the branch. Examiner will assume the former for this office action, but the language in the claims needs to be made more clear in the future, as the presence of the second "and" (and with the absence of commas) calls into question if prediction is a part of the group of things being prevented or if it is separate.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-8, 10-18, 20-28, and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Check et al. (USPN 6,125,444, herein Check), in view of Patterson et al. (herein Patterson).

5. As per Claim 1, Check teaches: A method operating a computer having a pipelined processor (Figure 1), comprising setting a bit within an instruction text field of a branch (Column 2, Lines 33-40 and Column 4, Lines 24-27), but fails to teach:

said bit preventing the branch from being placed into a branch history buffer and a branch target buffer to thereby prevent the branch from being written into the branch history buffer and branch target buffer and predicted and to make the branch only detectable as the time frame of decode.

While Check teaches using an instruction field of an instruction to disable a branch history table, he does not teach that doing so would prevent the branch from being placed in a branch target buffer, because Check is silent towards a branch target buffer even existing in the system. However, Patterson teaches that in order to further increase the performance of branches, a "branch target buffer" is often used, so that the target address can be calculated in the fetch stage, instead of the decode stage (Pages 271-275). As the branch target buffer relies on having a prediction in order to determine the correct address, as seen by Figure 4.23 on Page 275, if no prediction was able to be generated, because of a BHT being disabled, then it would have been obvious to one of ordinary skill in the art to also not use the BTB, as it would not have the data it needs to be made use of, therefore, attempting to use a BTB in this situation would not

only not make sense, but would be almost guaranteed to generate erroneous output. In addition, as can be seen by Column 2, Lines 28-31, sensitive system operations require cache control, so for the same reason Check disables the BHT, the BTB would need to not be written to as well. Given the advantage of a BTB as disclosed by Patterson, and the need to implement it in the system as disclosed by Check, one of ordinary skill in the art at the time the invention was made would have been motivated to include a BTB, and also to disable its use when the BHT was disabled, as the entire branch prediction mechanism must be disabled.

6. As per Claim 2, Check teaches: The method as defined in claim 1 comprising predicting the direction and target of a branch prior to decode (Figure 1).

7. As per Claim 3, Check teaches: A method as defined in claim 2 comprising predicting the direction and target of a branch prior to decode through a branch prediction array (Figure 1, also see Column 4, Lines 32-49).

8. As per Claim 4, Check teaches: A method as defined in claim 1 comprising tracking the branch from the beginning of the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array (It is inherent that instructions in a pipeline are kept track of, they must exist until they are removed).

9. As per Claim 5, Check teaches: A method as defined in claim 1 comprising denoting the instruction text field as a non-writable branch into the BTB (Column 2, Lines 36-40).

10. As per Claim 6, Check teaches: The method as defined in claim 5 comprising denoting the instruction text field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked from being written to the BTB (Column 2, Lines 28-31).

11. As per Claim 7, Check teaches: The method as defined in claim 6 comprising predicting the branch via aliasing (Column 4, Lines 12-15).

12. As per Claim 8, Patterson teaches: The method as defined in claim 1 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer (All instructions can alter machine state, so if any instruction is in the BTB, machine state altering code lies within an address range spanning by tag bits in the BTB).

13. As per Claim 10, Check teaches: The method as defined in claim 8 comprising denoting state altering code in the system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area (Column 2, Lines 28-40).

14. As per Claim 11, Check teaches: A computer system having input, output, storage, and a pipelined processor (Figure 1), said processor adapted and configured to set a bit within an instruction text field of a branch (Column 2, Lines 33-40 and Column 4, Lines 24-27), but fails to teach:

said bit preventing the branch from being placed into a branch history table and a branch target buffer to thereby prevent the branch from being written into the branch history buffer and branch target buffer and predicted and make the branch only detectable as the time frame of decode.

While Check teaches using an instruction field of an instruction to disable a branch history table, he does not teach that doing so would prevent the branch from being placed in a branch target buffer, because Check is silent towards a branch target buffer even existing in the system. However, Patterson teaches that in order to further increase the performance of branches, a "branch target buffer" is often used, so that the target address can be calculated in the fetch stage, instead of the decode stage (Pages 271-275). As the branch target buffer relies on having a prediction in order to determine the correct address, as seen by Figure 4.23 on Page 275, if no prediction was able to be generated, because of a BHT being disabled, then it would have been obvious to one of ordinary skill in the art to also not use the BTB, as it would not have the data it needs to be made use of, therefore, attempting to use a BTB in this situation would not only not make sense, but would be almost guaranteed to generate erroneous output. In addition, as can be seen by Column 2, Lines 28-31, sensitive system operations require cache control, so for the same reason Check disables the BHT, the BTB would need to

not be written to as well. Given the advantage of a BTB as disclosed by Patterson, and the need to implement it in the system as disclosed by Check, one of ordinary skill in the art at the time the invention was made would have been motivated to include a BTB, and also to disable its use when the BHT was disabled, as the entire branch prediction mechanism must be disabled.

15. As per Claim 12, Check teaches: The computer system as defined in claim 11, said computer system adapted and configured to predict the direction and target of a branch prior to decode (Figure 1).

16. As per Claim 13, Check teaches: The computer system as defined in claim 12 said computer system adapted and configured to predict the direction and target of a branch prior to decode through a branch prediction array (Figure 1, also see Column 4, Lines 32-49).

17. As per Claim 14, Check teaches: The computer system as defined in claim 11, said computer system adapted and configured to track the branch from the beginning of the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array (It is inherent that instructions in a pipeline are kept track of, they must exist until they are removed).

18. As per Claim 15, Check teaches: The computer system as defined in claim 11 said computer system adapted and configured to denote the instruction text field as a non-writable branch into the BTB (Column 2, Lines 36-40).

19. As per Claim 16, Check teaches: The computer system as defined in claim 15 said computer system adapted and configured to denote the instruction field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked (Column 2, Lines 28-31).

20. As per Claim 17, Check teaches: The computer system as defined in claim 16 said computer system adapted and configured to denote the instruction text field in the non-system area, and to predict the branch may be predicted via aliasing (Column 4, Lines 12-15).

21. As per Claim 18, Patterson teaches: The computer system as defined in claim 11 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer (All instructions can alter machine state, so if any instruction is in the BTB, machine state altering code lies within an address range spanning by tag bits in the BTB).

22. As per Claim 20, Check teaches: The computer system as defined in claim 18 said computer system is adapted and configured to denote state altering code in the

system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area is prevented (Column 2, Lines 28-40).

23. As per Claim 21, Check teaches: A program product comprising a storage medium having computer readable program code, said program code for use in a computer system having input, output, storage, and a pipelined processor (Figure 1), said program code adapting and configuring the computer system to set a bit within an instruction text field of a branch (Column 2, Lines 33-40 and Column 4, Lines 24-27), but fails to teach:

Said bit preventing the branch from being placed into a branch history table and a branch target buffer to thereby prevent the branch from being written into the branch history buffer and branch target buffer and predicted and make the branch only detectable as the time frame of decode.

While Check teaches using an instruction field of an instruction to disable a branch history table, he does not teach that doing so would prevent the branch from being placed in a branch target buffer, because Check is silent towards a branch target buffer even existing in the system. However, Patterson teaches that in order to further increase the performance of branches, a "branch target buffer" is often used, so that the target address can be calculated in the fetch stage, instead of the decode stage (Pages 271-275). As the branch target buffer relies on having a prediction in order to determine the correct address, as seen by Figure 4.23 on Page 275, if no prediction was able to be generated, because of a BHT being disabled, then it would have been obvious to

one of ordinary skill in the art to also not use the BTB, as it would not have the data it needs to be made use of, therefore, attempting to use a BTB in this situation would not only not make sense, but would be almost guaranteed to generate erroneous output. In addition, as can be seen by Column 2, Lines 28-31, sensitive system operations require cache control, so for the same reason Check disables the BHT, the BTB would need to not be written to as well. Given the advantage of a BTB as disclosed by Patterson, and the need to implement it in the system as disclosed by Check, one of ordinary skill in the art at the time the invention was made would have been motivated to include a BTB, and also to disable its use when the BHT was disabled, as the entire branch prediction mechanism must be disabled.

24. As per Claim 22, Check teaches: The program product as defined in claim 21, said computer system adapted and configured to predict the direction and target of a branch prior to decode (Figure 1).

25. As per Claim 23, Check teaches: The program product as defined in claim 22 said computer system adapted and configured to predict the direction and target of a branch prior to decode through a branch prediction array (Figure 1, also see Column 4, Lines 32-49).

26. As per Claim 24, Check teaches: The program product as defined in claim 21, said computer system adapted and configured to track the branch from the beginning of

the pipe, decode, until the time frame that the given instruction is to be written into a branch prediction array (It is inherent that instructions in a pipeline are kept track of, they must exist until they are removed).

27. As per Claim 25, Check teaches: The program product as defined in claim 21 said computer system adapted and configured to denote the instruction text field as a non-writable branch into the BTB (Column 2, Lines 36-40).

28. As per Claim 26, Check teaches: The program product as defined in claim 25 said computer system adapted and configured to denote the instruction text field in the system area as a non-writable branch into the BTB in system whereby the branch is blocked from being written to the BTB (Column 2, Lines 28-31).

29. As per Claim 27, Check teaches: The program product as defined in claim 26 said computer system adapted and configured to denote the instruction text field in the non-system area, and predict the branch via aliasing (Column 4, Lines 12-15).

30. As per Claim 28, Patterson teaches: The program product as defined in claim 21 wherein machine state altering code lies within an address range spanned by branch tag bits of the branch target buffer (All instructions can alter machine state, so if any instruction is in the BTB, machine state altering code lies within an address range

spanning by tag bits in the BTB).

31. As per Claim 30, Check teaches: The program product as defined in claim 28 said computer system is adapted and configured to denote state altering code in the system area by a state bit within the BTB/BHT such that aliasing of branches is prevented within the system area (Column 2, Lines 28-40).

Response to Arguments

32. Applicants arguments essentially seem to boil down to 2 issues, that the rejection is a hindsight reconstruction, and that the combination of references fail to teach two limitations, which Applicant argues are conclusions. However, Examiner is not persuaded by these arguments. As stated in previous remarks, there is no hindsight reconstruction required to reject these claims, and Examiner asserts that his rejection, and the way the references are combined, are the only way that one of ordinary skill in the art could possibly come to. Check teaches why the branch history table should be disabled, for areas where the user explicitly does not want branch prediction to be used. Examiner can see absolutely no reason, in the situation where the user explicitly does not want branch prediction to occur, that the user would disable the BHT, but continue to use a BTB, a component that only exists for one purpose, to assist in branch prediction. To combine the references in the way the Applicant is arguing is proper, a user would have to disable the BHT and not use it, in order to avoid prediction, but still for some reason decide that he wants prediction to still be used and then access the

BTB. Additionally, because the BTB relies upon input from the branch prediction mechanisms, it could not even possibly be correctly used, and if one was to even attempt to use a BTB with the BHT disabled, the BTB would output erroneous or random data, causing the computer to execute random instructions. Examiner asserts that no one of ordinary skill in the art would ever come to this conclusion, and that the only thing that makes sense in the context of this invention is to disable all components associated with branch prediction when the user has explicitly indicated that branch prediction is to be disabled, and given that a BTB has no purpose outside of branch prediction, and that it cannot work with other branch prediction mechanisms disabled, that it would be disabled also.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert E. Fennema whose telephone number is (571) 272-2748. The examiner can normally be reached on Monday-Friday, 8:45-6:15.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Robert E Fennema
Examiner
Art Unit 2183

RF

RICHARD L. ELLIS
PRIMARY EXAMINER